CPP 程式設計題

命題者：Shen

題目名稱(中文/英文)：Don't starve yourself!

主要測試觀念：Class , String, Function

Basics	Functions
☐ C++ BASICS	☐ SEPARATE COMPILATION AND NAMESPACES
■ FLOW OF CONTROL	☐ STREAMS AND FILE I/O
■ FUNCTION BASICS	☐ RECURSION
☐ PARAMETERS AND OVERLOADING	☐ INHERITANCE
☐ ARRAYS	☐ POLYMORPHISM AND VIRTUAL FUNCTIONS
■ STRUCTURES AND CLASSES	☐ TEMPLATES
■ CONSTRUCTORS AND OTHER TOOLS	☐ LINKED DATA STRUCTURES
☐ OPERATOR OVERLOADING, FRIENDS, AND REFERENCES	☐ EXCEPTION HANDLING
■ STRINGS	☐ STANDARD TEMPLATE LIBRARY
☐ POINTERS AND DYNAMIC ARRAYS	☐ PATTERNS AND UML

題目說明：

Xiao Ming likes to read adventure story. He wants to record the character's state through the adventure. In the adventure, character will eat whatever food he/she meets.

What you need to do:

1. Implement two classes **Food** and **Character** to record some information.
2. **Character** should have three properties: **satisfaction(滿足度), sanity(理智值) and name**. **Food** should have two properties **satisfaction and sanity**, which means what your character eat will affect character's satisfaction and sanity.
3. You also need to implement a function **feast(Food)** in the character class, to store the effect of the food. Don't forget to implement a function **show()** in the character class to show the character's state.
   After all of this, you can start your adventure by typing some content.
4. There will be 3 kinds of food for now: **jerky(meat), carrot(vegetable), monster meat(bad meat)**, each kind would have different effect.(I already define these for you in the main.cpp) You need to detect whether they exist in your content or not. Character will eat whatever food he/she meets, so you only need identify word but not verb. Your adventure stops when you type **"end"**.

輸入說明：

1. First line should be your character's **name**. (No more than 9 words)
2. Then, type in the content of your adventure. Update character's state each line **(updated by the function feast(Food))**. **Show your character's state every line**.
3. We only recognize the food with **all lowercase letters**. (jerky but not Jerky)
4. And when you type in a single **"end" or "End"**, finish the adventure.

Some **special case** you need to notice will be discussed below.**(其他註記區）**

輸出說明：

1. Output format is in output.txt.
2. You only need to print the result on the screen.(Don't output a file.)

IO 範例:	
Sample Input	Sample Output (whole version is in output.txt)
Wilson He finds some jerky. He finds some carrot. He picks up some monster meat. He gets jerky and carrot. At the end, he finds a jerky again! End	Please see the output.txt

附屬資料:

☑解答程式:

☑測試資料:output.txt

■易,僅需用到基礎程式設計語法與結構

□中,需用到多項程式設計語法與結構

□難,需用到多項程式結構或較為複雜之資料型態或結構

解題時間:25 分鐘

其他註記:

1. When there are more than two kinds of food in one line, you should consider both of their effect.

I pick up jerky and monster meat. → **Consider both of them!**

2. If there are same food in one line, you should consider it only once.

Ex: I eat two jerky. → **You still consider it once.**

3. An adventure only ends with a single "end" word.

Ex: At the end, I have a jerky. → **Won't end the adventure!**

Ex: end. → **The adventure ends.**

4. Hints: setw(10) for name, setw(20) for satisfaction and sanity. Please make sure about your alignment. If we can't figure your answer, you can't get the score.